ELSEVIER

Journal of Computational and Applied Mathematics 108 (1999) 57–73

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS



# Integrable discretization and its applications

Norimasa Shibutani

*Department of Mathematics, Kurume Institute of Technology, Kurume, Fukuoka 830-0052, Japan*

Received 1 October 1998; received in revised form 25 March 1999

**Abstract**

Using an integrable discretization of the Rayleigh quotient system, a new algorithm for computing the largest eigenvalue is obtained. The Rayleigh quotient system is discretized by our own method (Bull. Kurume Inst. Technol 11 (1987) 1–7), which solves a quadratic differential equation explicitly. The algorithm converges more rapidly than Wilkinson's power method with a shift of the origin. © 1999 Published by Elsevier Science B.V. All rights reserved.

*MSC:* 65F15; 15A18; 58F07; 39A10

*Keywords:* Integrable discretization; Rayleigh quotient; Power method with a shift

## 1. Introduction

Recently, Nakamura et al. [3] introduced the terminology *integrable discretization*, and suggested that integrable discretization will be useful in designing numerical algorithms. Integrable discretization involves the discretization of nonlinear integrable systems in a linear level, an idea which was also developed by Hirota [2] in soliton equations. Nakamura et al. [3] showed that the discrete Rayleigh quotient system is essentially equivalent to the power method with a shift of the origin. Nakamura [4] also described an equivalence between the Jacobi algorithm and Lax form. In this paper we discretize the Rayleigh quotient system by our own method [5], which is different to that of [3], and obtain a new algorithm for computing the largest eigenvalue.

Let $A$ be an $N \times N$ real symmetric matrix having eigenvalues such that $\lambda_1 > \lambda_2 \geqslant \cdots \geqslant \lambda_{N-1} > \lambda_N$. The power method with a shift of the origin for computing the largest eigenvalue is known as Wilkinson's method (see [6, p. 572, 1]). That is, the iteration is as follows:

$$\boldsymbol{x}(n+1) = \frac{(I + \varepsilon A)\boldsymbol{x}(n)}{\|(I + \varepsilon A)\boldsymbol{x}(n)\|}, \quad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{1}$$

*E-mail address:* sibutani@cc.kurume-it.ac.jp (N. Shibutani)


PII: S0377-0427(99)00100-4

where $I$ is an $N \times N$ identity matrix, $\boldsymbol{x}$ a real $N$-vector, and $\varepsilon$ is a real number. Iteration (1) leads to the power method as $|\varepsilon| \to \infty$. For the convergence to the eigenvector $\boldsymbol{x}_1$ corresponding to $\lambda_1$, the optimum value of $\varepsilon$ is given by

$$\varepsilon_{\text{opt}} = -\frac{2}{\lambda_2 + \lambda_N}$$

and the convergence is governed by the quantity

$$\left(\frac{\lambda_2 - \lambda_N}{2\lambda_1 - \lambda_2 - \lambda_N}\right)^n.$$

In this paper, the following algorithm is introduced, in which for convenience we suppose that $\lambda_N = 0$:

$$\boldsymbol{x}(n+1) = \frac{(A - \varepsilon_0 A^2)\boldsymbol{x}(n)}{\|(A - \varepsilon_0 A^2)\boldsymbol{x}(n)\|}, \quad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2}$$

where

$$\varepsilon_0 = \frac{1 + \sqrt{2}}{2\lambda_2}.$$

We will show that the convergence for iteration (2) is more rapid than Wilkinson's method. Let $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$ be the orthonormal eigenvectors corresponding to $\lambda_1, \lambda_2, \ldots, \lambda_N$, respectively. If we write $\boldsymbol{x}_0 = \sum_{j=1}^{N} \alpha_j \boldsymbol{x}_j$, then apart from the normalizing factor, $\boldsymbol{x}(n)$ is given by

$$\sum_{j=1}^{N} \alpha_j (\lambda_j - \varepsilon_0 \lambda_j^2)^n \boldsymbol{x}_j.$$

We assume throughout this paper that $\alpha_k \neq 0$ when we discuss the convergence to the eigenvector $\boldsymbol{x}_k$. If we use the notation $f(x, a) = a - a^2 x$ and use $\varepsilon$ instead of $\varepsilon_0$, then it becomes

$$\sum_{j=1}^{N} \alpha_j (f(\varepsilon, \lambda_j))^n \boldsymbol{x}_j = (f(\varepsilon, \lambda_1))^n \left\{ \alpha_1 \boldsymbol{x}_1 + \sum_{j=2}^{N} (f(\varepsilon, \lambda_j)/f(\varepsilon, \lambda_1))^n \alpha_j \boldsymbol{x}_j \right\}.$$

If

$$\max_{j \neq 1} \left| \frac{f(\varepsilon, \lambda_j)}{f(\varepsilon, \lambda_1)} \right| < 1, \tag{3}$$

then $\boldsymbol{x}(n)$ converges to the eigenvector $\boldsymbol{x}_1$ as $n \to \infty$. Here, a remark is in order. If $f(\varepsilon, \lambda_1) < 0$ and $\alpha_1 > 0$, then $\boldsymbol{x}(2n)$ tends to $\boldsymbol{x}_1$ and $\boldsymbol{x}(2n+1)$ to $-\boldsymbol{x}_1$, respectively, as $n \to \infty$, nevertheless $\langle \boldsymbol{x}(n), A\boldsymbol{x}(n) \rangle$ converges to $\lambda_1$ as $n \to \infty$, where $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^{\mathrm{T}} \boldsymbol{y}$ for real $N$-vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. In this context we simply say that $\boldsymbol{x}(n)$ converges to $\boldsymbol{x}_1$. Also, the sign of $f(x, a)$ is not important. We call the left-hand side quantity in (3) the rate of convergence. The optimum value $\varepsilon_{\text{opt}}$ is characterized as follows:

$$\max_{j \neq 1} \left| \frac{f(\varepsilon_{\text{opt}}, \lambda_j)}{f(\varepsilon_{\text{opt}}, \lambda_1)} \right| = \min_{\varepsilon} \max_{j \neq 1} \left| \frac{f(\varepsilon, \lambda_j)}{f(\varepsilon, \lambda_1)} \right|. \tag{4}$$

We can consider other functions of $f(x,a)$, for example, $f(x,a)=1+ax$ for Wilkinson's method, or $f(x,a)=a^2x^2-2ax-2$ for second-order Runge–Kutta method in Section 5.

For the present case it is easy to show that

$$\max_{j\neq 1}\left|\frac{f(\varepsilon_0,\lambda_j)}{f(\varepsilon_0,\lambda_1)}\right|=\left|\frac{f(\varepsilon_0,\lambda_2)}{f(\varepsilon_0,\lambda_1)}\right|=\frac{1}{(3+2\sqrt{2})r^2-2(1+\sqrt{2})r}<\frac{1}{(2r-1)^2}$$

for $r\equiv\lambda_1/\lambda_2>1$. The left-hand side of the inequality coincides with the rate of our algorithm and the right-hand side is the square of the rate of Wilkinson's method. That is, a single step of our method converges faster than two steps of Wilkinson's method. But the computational requirement of each step in the two methods is the same because we do not compute $A^2$. In the case of $\lambda_N\neq 0$, we can apply this algorithm to $A-\lambda_N I$ instead of $A$.

Our purpose in this paper is to describe how to find algorithms like the one above. The integrable discretization enables us to realize this purpose most effectively.

## 2. Preliminaries

The starting point in the derivation of integrable differential equations is to consider the Rayleigh quotient of $A$,

$$R_A(\boldsymbol{x})\equiv\frac{\langle\boldsymbol{x},A\boldsymbol{x}\rangle}{\langle\boldsymbol{x},\boldsymbol{x}\rangle}.$$

The well-known minimax theorem states that

$$\lambda_1=\max_{\|\boldsymbol{x}\|=1}R_A(\boldsymbol{x}),\qquad \lambda_N=\min_{\|\boldsymbol{x}\|=1}R_A(\boldsymbol{x}),$$

where $\|\boldsymbol{x}\|^2=\langle\boldsymbol{x},\boldsymbol{x}\rangle$. One of the simplest strategies for maximizing $R_A(\boldsymbol{x})$ is the method of steepest ascent. At a current point $\boldsymbol{x}$, where $\|\boldsymbol{x}\|=1$, the function $R_A(\boldsymbol{x})$ increases most rapidly in the direction of the positive gradient: $\nabla R_A(\boldsymbol{x})=2A\boldsymbol{x}-2\langle\boldsymbol{x},A\boldsymbol{x}\rangle\boldsymbol{x}$, which is restricted on the unit sphere. Thus the maximal eigenvalue $\lambda_1$ of $A$ can be calculated through the trajectory of the gradient system:

$$\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t}=A\boldsymbol{x}-\langle\boldsymbol{x},A\boldsymbol{x}\rangle\boldsymbol{x},\quad \boldsymbol{x}(0)=\boldsymbol{x}_0,\ \|\boldsymbol{x}_0\|=1. \tag{5}$$

Also, for minimizing $R_A(\boldsymbol{x})$ the negative gradient will be used. In other words we use $-A$ instead of $A$, or we may use the same Eq. (5) for negative time. Note that if $\|\boldsymbol{x}_0\|=1$, then $\|\boldsymbol{x}(t)\|=1$. Even if we discretize (5) by Euler's method, we still cannot have a conservative quantity. As we see later, the nonlinear equation (5) is essentially linear, if we discretize the linear equation, then the discretization may have a conservative quantity. This is a fundamental technique in the integrable discretization (cf. [2,3]).

Let $P$ be an orthogonal matrix which diagonalizes $A$ as $D\equiv P^{\mathrm{T}}AP=\mathrm{diag}(\lambda_1,\ldots,\lambda_N)$. Using $\boldsymbol{x}=P\boldsymbol{r}$, $D=P^{\mathrm{T}}AP$, Eq. (5) can be transformed into the following equivalent form:

$$\frac{\mathrm{d}r_j^2}{\mathrm{d}t}=2\lambda_j r_j^2-2r_j^2\sum_{k=1}^{N}\lambda_k r_k^2, \tag{6}$$

where $\boldsymbol{r}=(r_1,\ldots,r_N)^{\mathrm{T}}$. Let $\boldsymbol{y}\equiv(r_1^2,\ldots,r_N^2)^{\mathrm{T}}$, $\boldsymbol{d}\equiv(\lambda_1,\ldots,\lambda_N)^{\mathrm{T}}$. Then the gradient system (5) is also equivalent to the following form:

$$\frac{\mathrm{d}\boldsymbol{y}(t)}{\mathrm{d}t}=2D\boldsymbol{y}-2\langle\boldsymbol{d},\boldsymbol{y}\rangle\boldsymbol{y},\quad \boldsymbol{y}(0)=\boldsymbol{y}_0, \tag{7}$$

where $\boldsymbol{y}_0=(r_1^2(0),\ldots,r_N^2(0))^{\mathrm{T}}$.

In order to solve Eq. (7), we use the method in Shibutani [5] described below. Letting $A$ be an $N\times N$ real matrix, not necessarily symmetric, we consider the following system of differential equations:

$$\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t}=\boldsymbol{a}+A\boldsymbol{x}+\langle\boldsymbol{b},\boldsymbol{x}\rangle\boldsymbol{x},\quad \boldsymbol{x}(0)=\boldsymbol{x}_0, \tag{8}$$

where $\boldsymbol{a}$ and $\boldsymbol{b}$ are constant vectors. We can solve this system as follows:

(i) Solve the following system of linear differential equations for $\hat{\boldsymbol{x}},\hat{u}$ with explicit form in $\boldsymbol{x}$ and $u$:

$$\frac{\mathrm{d}\hat{\boldsymbol{x}}}{\mathrm{d}t}=-A^{\mathrm{T}}\hat{\boldsymbol{x}}-\hat{u}\boldsymbol{b},\quad \frac{\mathrm{d}\hat{u}}{\mathrm{d}t}=\langle\boldsymbol{a},\hat{\boldsymbol{x}}\rangle,\quad \hat{\boldsymbol{x}}(0)=\boldsymbol{x},\ \hat{u}(0)=u, \tag{9}$$

where $\boldsymbol{x}$ is a constant vector and $u$ is a real number.

(ii) Solve the following simultaneous equations $\hat{\boldsymbol{x}}=\hat{\boldsymbol{x}}(t,\boldsymbol{x},u)$, $\hat{u}=\hat{u}(t,\boldsymbol{x},u)$ for $\boldsymbol{x},u$ and substitute them into $u=\langle\boldsymbol{x},\boldsymbol{x}_0\rangle$.

(iii) By taking the gradient of $\hat{u}$ with respect to $\hat{\boldsymbol{x}}$, we obtain the solution $\boldsymbol{x}(t)\equiv\nabla\hat{u}(t)$ of Eq. (8).

**Example 1.** As the simplest example of our method we solve the logistic differential equation:

$$\frac{\mathrm{d}x}{\mathrm{d}t}=ax(1-x),\quad x(0)=x_0, \tag{10}$$

where $a$ is a positive constant. The solutions of the linear differential equation,

$$\frac{\mathrm{d}\hat{x}}{\mathrm{d}t}=-a\hat{x}+a\hat{u},\quad \frac{\mathrm{d}\hat{u}}{\mathrm{d}t}=0,\quad \hat{x}(0)=x,\ \hat{u}(0)=u, \tag{11}$$

are $\hat{x}(t)=(x-u)\mathrm{e}^{-at}+u$, $\hat{u}(t)=u$. Solve the equations $\hat{x}=(x-u)\mathrm{e}^{-at}+u$, $\hat{u}=u$ for $x,u$, then we have $x=\hat{u}+(\hat{x}-\hat{u})\mathrm{e}^{at}$, $u=\hat{u}$. Substitute them into $u=xx_0$, and find $\mathrm{d}\hat{u}/\mathrm{d}\hat{x}$, then the solution obtained is

$$x(t)=\frac{\mathrm{e}^{at}x_0}{1+(\mathrm{e}^{at}-1)x_0}. \tag{12}$$

## 3. Exact solution

In order to solve gradient system (5) explicitly, we apply the above method in Shibutani [5] to Eq. (7). It leads to the following linear differential equation:

$$\frac{\mathrm{d}\hat{\boldsymbol{y}}(t)}{\mathrm{d}t}=-2D\hat{\boldsymbol{y}}+2\hat{u}\boldsymbol{d},\quad \hat{\boldsymbol{y}}(0)=\boldsymbol{y}. \tag{13}$$

The solution of this linear system is given by

$$\hat{\boldsymbol{y}} = \mathrm{e}^{-2tD}(\boldsymbol{y} - \hat{u}\boldsymbol{e}) + \hat{u}\boldsymbol{e}, \tag{14}$$

where $\boldsymbol{e} \equiv (1, 1, \ldots, 1)^{\mathrm{T}}$. Solve (14) for $\boldsymbol{y}$ and substitute it into $\hat{u} = \langle \boldsymbol{y}, \boldsymbol{y}_0 \rangle$ where $\boldsymbol{y}_0$ is defined in (7). By taking the gradient of $\hat{u}$ with respect to $\hat{\boldsymbol{y}}$, it follows that

$$\nabla \hat{u} = \mathrm{e}^{2tD}\boldsymbol{y}_0 - \langle \mathrm{e}^{2tD}\boldsymbol{e}, \boldsymbol{y}_0 \rangle \nabla \hat{u} + \langle \boldsymbol{e}, \boldsymbol{y}_0 \rangle \nabla \hat{u}.$$

Using the relation $\langle \boldsymbol{e}, \boldsymbol{y}_0 \rangle = \sum_{k=1}^{N} y_k(0) = \| \boldsymbol{r}_0 \|^2 = 1$, we obtain the following solution of Eq. (7):

$$\boldsymbol{y}(t) = \nabla \hat{u} = \frac{\mathrm{e}^{2tD}\boldsymbol{y}_0}{\langle \mathrm{e}^{2tD}\boldsymbol{e}, \boldsymbol{y}_0 \rangle}.$$

Hence, it follows that

$$\boldsymbol{y}(2t) = \frac{\mathrm{e}^{4tD}\boldsymbol{y}_0}{\langle \mathrm{e}^{4tD}\boldsymbol{e}, \boldsymbol{y}_0 \rangle} = \frac{\mathrm{e}^{4tD}\boldsymbol{y}_0}{\| \mathrm{e}^{2tD}\boldsymbol{r}(0) \|^2}.$$

Since $y_j = r_j^2$, we have

$$\boldsymbol{r}(2t) = \frac{\mathrm{e}^{2tD}\boldsymbol{r}(0)}{\| \mathrm{e}^{2tD}\boldsymbol{r}(0) \|},$$

where the sign is uniquely determined by the initial condition. Using again $\boldsymbol{r} = P^{\mathrm{T}}\boldsymbol{x}$ and $PDP^{\mathrm{T}} = A$, we obtain the following exact solution of (5):

$$\boldsymbol{x}(t) = \frac{\mathrm{e}^{tA}\boldsymbol{x}_0}{\| \mathrm{e}^{tA}\boldsymbol{x}_0 \|}. \tag{15}$$

Next, we will discuss the convergence of the exact solution as $t \to \infty$ and the error estimate. If we use $\boldsymbol{x}_0 = \sum \alpha_j \boldsymbol{x}_j$, then apart from the normalizing factor, exact solution (15) is given by

$$\sum_{j=1}^{N} \alpha_j \mathrm{e}^{\lambda_j t}\boldsymbol{x}_j = \mathrm{e}^{\lambda_1 t}\left\{ \alpha_1 \boldsymbol{x}_1 + \sum_{j=2}^{N} \mathrm{e}^{(\lambda_j - \lambda_1)t}\alpha_j \boldsymbol{x}_j \right\}.$$

Therefore, as $t \to \infty$, $\boldsymbol{x}(t)$ converges to the eigenvector corresponding to $\lambda_1$. As $t \to -\infty$, $\boldsymbol{x}(t)$ converges to the eigenvector corresponding to $\lambda_N$.

**Theorem 1.** *Let $A$ be an $N \times N$ real symmetric matrix having eigenvalues such that $\lambda_1 > \lambda_2 \geqslant \cdots \geqslant \lambda_N$. Then, for the solution $\boldsymbol{x}(t)$ of Eq. (5), the error estimate for the maximal eigenvalue is stated as*

$$0 \leqslant \lambda_1 - \langle \boldsymbol{x}(t), A\boldsymbol{x}(t) \rangle \leqslant K\mathrm{e}^{-2t(\lambda_1 - \lambda_2)} \quad \text{for } t \geqslant 0 \tag{16}$$

*for some $K > 0$.*

**Proof.** We have

$$\lambda_1 - \langle \boldsymbol{x}(t), A\boldsymbol{x}(t) \rangle = \lambda_1 - \langle \boldsymbol{r}(t), D\boldsymbol{r}(t) \rangle$$

$$= \lambda_1 - \frac{\langle \mathrm{e}^{tD}\boldsymbol{r}(0), D\mathrm{e}^{tD}\boldsymbol{r}(0) \rangle}{\| \mathrm{e}^{tD}\boldsymbol{r}(0) \|^2} = \frac{\sum_{j=2}^{N} (\lambda_1 - \lambda_j)\mathrm{e}^{2t\lambda_j}(r_j(0))^2}{\sum_{j=1}^{N} \mathrm{e}^{2t\lambda_j}(r_j(0))^2}.$$

From this, it follows that $0 \leqslant \lambda_1 - \langle \boldsymbol{x}(t), A\boldsymbol{x}(t)\rangle$. Also we have

$$\frac{\sum_{j=2}^{N}(\lambda_1-\lambda_j)\mathrm{e}^{2t(\lambda_j-\lambda_1)}(r_j(0))^2}{\sum_{j=2}^{N}\mathrm{e}^{2t(\lambda_j-\lambda_1)}(r_j(0))^2+(r_1(0))^2} \leqslant \frac{\mathrm{e}^{-2t(\lambda_1-\lambda_2)}}{(r_1(0))^2}\sum_{j=2}^{N}(\lambda_1-\lambda_j)(r_j(0))^2$$

$$\leqslant K\mathrm{e}^{-2t(\lambda_1-\lambda_2)}. \qquad \square$$

**Corollary 1.** *Let $A$ be an $N\times N$ real symmetric matrix having eigenvalues such that $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_{N-1} > \lambda_N$. Then the error estimate for the minimal eigenvalue is stated as*

$$0 \leqslant \langle \boldsymbol{x}(t), A\boldsymbol{x}(t)\rangle - \lambda_N \leqslant K_1\mathrm{e}^{-2t(\lambda_N-\lambda_{N-1})} \quad \text{for } t\leqslant 0 \tag{17}$$

*for some $K_1 > 0$.*

## 4. Integrable discretization

In this section it will be shown that if we discretize linear equation (13) using the forward Euler method, then the inverse iteration (also called fractional iteration) algorithm is derived. If we use the backward Euler method, then the power method with a shift of the origin is obtained.

Consider the following approximation of (13) by the forward Euler method:

$$\hat{\boldsymbol{y}}(n+1) - \hat{\boldsymbol{y}}(n) = 2\varepsilon(-D\hat{\boldsymbol{y}}(n) + \hat{u}\boldsymbol{d}), \tag{18}$$

where $\varepsilon$ is a stepsize and we denote $\hat{\boldsymbol{y}}(\varepsilon n)$ by $\hat{\boldsymbol{y}}(n)$. It suffices to solve for $\boldsymbol{y}(1)$ only, so we have

$$\hat{\boldsymbol{y}}(1) = (I - 2\varepsilon D)\boldsymbol{y} + 2\varepsilon\hat{u}\boldsymbol{d}. \tag{19}$$

Solve (19) for $\boldsymbol{y}$ and substitute it into $\hat{u} = \langle \boldsymbol{y}, \boldsymbol{y}_0\rangle$. By taking the gradient of $\hat{u}$ with respect to $\hat{\boldsymbol{y}}(1)$, we obtain the solution

$$\boldsymbol{y}(1) = \nabla\hat{u} = \frac{(I-2\varepsilon D)^{-1}\boldsymbol{y}_0}{1+2\varepsilon\langle (I-2\varepsilon D)^{-1}\boldsymbol{d}, \boldsymbol{y}_0\rangle}.$$

For each component, we have, by the fact that $\sum_{k=1}^{N} y_k(0) = 1$,

$$y_j(1) = \frac{y_j(0)/(1-2\varepsilon\lambda_j)}{1+2\varepsilon\sum_{k=1}^{N}\lambda_k y_k(0)/(1-2\varepsilon\lambda_k)} = \frac{y_j(0)/(1-2\varepsilon\lambda_j)}{\sum_{k=1}^{N} y_k(0)/(1-2\varepsilon\lambda_k)}.$$

This formula shows $\sum_{k=1}^{N} y_k(1) = 1$. Therefore, we have

$$y_j(2) = \frac{y_j(0)/(1-2\varepsilon\lambda_j)^2}{\sum_{k=1}^{N} y_k(0)/(1-2\varepsilon\lambda_k)^2}. \tag{20}$$

Since $y_j(n) = r_j^2(n)$, we have

$$\boldsymbol{r}(2) = \frac{(I-2\varepsilon D)^{-1}\boldsymbol{r}(0)}{\|(I-2\varepsilon D)^{-1}\boldsymbol{r}(0)\|}, \tag{21}$$

where the sign is uniquely determined by the condition that $\boldsymbol{r}(2) \to \boldsymbol{r}(0)$ as $\varepsilon \to 0$. In general, using $\boldsymbol{r} = P^{\mathrm{T}}\boldsymbol{x}$, $PDP^{\mathrm{T}} = A$, we have

$$\boldsymbol{x}(n+2) = \frac{(I - 2\varepsilon A)^{-1}\boldsymbol{x}(n)}{\|(I - 2\varepsilon A)^{-1}\boldsymbol{x}(n)\|}. \tag{22}$$

Thus the inverse iteration is obtained.

In a similar way, if we discretize Eq. (13) by using the backward Euler method, then the following difference equation is obtained:

$$\boldsymbol{x}(n+1) = \frac{(I + \varepsilon A)\boldsymbol{x}(n)}{\|(I + \varepsilon A)\boldsymbol{x}(n)\|}, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{23}$$

This algorithm is the power method with a shift of the origin.

Next, we will discretize Eq. (13) by the second-order backward Runge–Kutta method. Then we have

$$\hat{\boldsymbol{y}}(1) - \hat{\boldsymbol{y}}(0) = (\boldsymbol{k}_1 + \boldsymbol{k}_2)/2, \tag{24}$$

where $\boldsymbol{F}(\hat{\boldsymbol{y}}) \equiv -D\hat{\boldsymbol{y}} + \hat{u}\boldsymbol{d}$, $\boldsymbol{k}_1 \equiv \varepsilon\boldsymbol{F}(\hat{\boldsymbol{y}}(1))$ and $\boldsymbol{k}_2 \equiv \varepsilon\boldsymbol{F}(\hat{\boldsymbol{y}}(1) + \boldsymbol{k}_1)$. Hence it follows that

$$\boldsymbol{y} = \hat{\boldsymbol{y}}(1) - \frac{1}{2}(\boldsymbol{k}_1 + \boldsymbol{k}_2) = \left(I + \varepsilon D - \frac{\varepsilon^2}{2}D^2\right)\hat{\boldsymbol{y}}(1) - \hat{u}\left(\varepsilon I - \frac{\varepsilon^2}{2}D\right)\boldsymbol{d}.$$

Substitute $\boldsymbol{y}$ into $\hat{u} = \langle \boldsymbol{y}, \boldsymbol{y}_0 \rangle$. By taking the gradient of $\hat{u}$ with respect to $\hat{\boldsymbol{y}}(1)$, we have

$$\boldsymbol{y}(1) = \nabla\hat{u} = \frac{(I + \varepsilon D - (\varepsilon^2/2)D^2)\boldsymbol{y}_0}{1 + \langle(\varepsilon I - (\varepsilon^2/2)D)\boldsymbol{d}, \boldsymbol{y}_0\rangle}.$$

In general, by using $\sum_{k=1}^{N} y_k(n) = 1$, it follows that

$$y_j(n+1) = \frac{(1 + \varepsilon\lambda_j - \varepsilon^2\lambda_j^2/2)y_j(n)}{\sum_{k=1}^{N}(1 + \varepsilon\lambda_k - \varepsilon^2\lambda_k^2/2)y_k(n)}.$$

Thus we have the following difference equation:

$$\boldsymbol{x}(n+1) = \frac{(I + \varepsilon A - (\varepsilon^2/2)A^2)\boldsymbol{x}(n)}{\|(I + \varepsilon A - (\varepsilon^2/2)A^2)\boldsymbol{x}(n)\|}, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{25}$$

**Remark 1.** For the logistic differential equation if we discretize (11) by the backward Euler method, then we obtain

$$x(n) = \frac{(1 + \varepsilon a)^n x_0}{1 + ((1 + \varepsilon a)^n - 1)x_0}.$$

This difference solution shows that chaos does not occur.

## 5. Discretization by second-order backward Runge–Kutta method

In this section we discuss the following difference equation, which is discretized by the second-order backward Runge–Kutta method to Eq. (7),

$$\boldsymbol{x}(n+1) = \frac{(I + \varepsilon A - (\varepsilon^2/2)A^2)\boldsymbol{x}(n)}{\|(I + \varepsilon A - (\varepsilon^2/2)A^2)\boldsymbol{x}(n)\|}, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{26}$$

For fixed $t = n\varepsilon$, $\boldsymbol{x}(n)$ tends to the exact solution $\boldsymbol{x}(t)$ of (5) as $n \to \infty$, because $\lim_{n\to\infty}(1 + \varepsilon\lambda_j - \varepsilon^2\lambda_j^2/2)^n = \mathrm{e}^{\lambda_j t}$ for each $j$ ($j = 1, 2, \ldots, N$). Since $\varepsilon_{\mathrm{opt}}$ is not so small in Wilkinson's method, we may consider the difference equation (26) for $\varepsilon$ which is not necessarily small. Thus the integrable discretization gives us an algorithm for eigenvalue. Throughout this section we set

$$f(x, a) = a^2x^2 - 2ax - 2. \tag{27}$$

**Definition 1.** Let $\lambda_1 > \lambda_2 > \cdots > \lambda_{N-1} > \lambda_N \geqslant 0$. For $i \neq j$, we denote the two roots of the equation $f(x, \lambda_i) = -f(x, \lambda_j)$ by $\alpha_j^i < 0 < \beta_j^i$. We define $x_p \equiv \beta_p^1 = \max_{2\leqslant j\leqslant N}\{\beta_j^1\}$, where if there exist two integers $p$ and $p'$ such that $\beta_p^1 = \beta_{p'}^1$, then we adopt the larger one. Moreover for $j = 1, 2, \ldots, N-1$, we define $\ell_j$ as the positive root of $f(x, \lambda_j) = f(x, \lambda_{j+1})$, and set $\ell_0 = 0$, $\ell_N = +\infty$. Finally, let $\ell_-$ be the negative root of $f(x, \lambda_1) = -f(x, \lambda_N)$.

**Lemma 1.** *Let $x_p$ be as defined in Definition* 1. *Then it holds that $\ell_{p-1} \leqslant x_p < \ell_p$. Also for each $j = 2, 3, \ldots, p-1$, $\ell_j \leqslant \beta_j^1$.*

**Proof.** Notice that $\ell_j = 2/(\lambda_j + \lambda_{j+1})$ for $j = 1, 2, \ldots, N-1$. For each $j = 1, 2, \ldots, N$, it holds that

$$\text{if } k \neq j \text{ then} \quad f(x, \lambda_j) < f(x, \lambda_k) \quad \text{for } \ell_{j-1} < x < \ell_j. \tag{28}$$

For $x \geqslant 0$ we define $h(x)$ as follows: $h(x) = -f(x, \lambda_j)$ if $\ell_{j-1} \leqslant x < \ell_j$ ($j = 1, 2, \ldots, N$). Then the equation $f(x, \lambda_1) = h(x)$ has a unique solution denoted by $\beta$. Since $-f(x, \lambda_j) \leqslant h(x)$ and $f(x, \lambda_1)$ is a strictly monotone increasing function for $x > 1/\lambda_1$, it holds that $\beta_j^1 \leqslant \beta$, that is, $\beta = x_p = \beta_p^1$. Since $\ell_{p-1} \leqslant \beta_p^1 \leqslant \ell_p$, and by the definition of $x_p$, that is, $p$ is the larger one, it follows that $\ell_{p-1} \leqslant x_p < \ell_p$. The second part of Lemma 1 is proved as well. In general, for $c > a > b \geqslant 0$, we denote the positive roots of $f(x, c) = -f(x, a)$, $f(x, c) = -f(x, b)$ and $f(x, a) = f(x, b)$ by $\alpha$, $\beta$ and $\ell$, respectively. Then there is a possibility of following three cases only, that is, $\ell < \alpha < \beta$, $\beta < \alpha < \ell$ or $\alpha = \beta = \ell$. Applying these formulas for $c = \lambda_1$, $a = \lambda_j$ ($j = 2, 3, \ldots, p-1$), and $b = \lambda_p$, by considering $2/(\lambda_j + \lambda_p) \leqslant \ell_{p-1} \leqslant x_p = \beta_p^1$, it follows that $2/(\lambda_j + \lambda_p) \leqslant \beta_j^1$. Therefore, we have $2/(\lambda_j + \lambda_{j+1}) \leqslant \beta_j^1$ ($j = 2, 3, \ldots, p-1$). □

**Theorem 2.** *Let $A$ be an $N \times N$ real symmetric matrix having eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_{N-1} > \lambda_N \geqslant 0$. Let $x_p$ be as defined in Definition* 1. *Then the convergence property of $\{\boldsymbol{x}(n)\}_{n=0,1,2,\ldots}$ defined by* (26) *is as follows*:

(i) *For $\varepsilon < \ell_-$, $0 < \varepsilon < \ell_1$, $x_p < \varepsilon$, $\boldsymbol{x}(n)$ tends to the eigenvector $\boldsymbol{x}_1$.*
(ii) *For $\ell_{j-1} < \varepsilon < \ell_j$, ($j = 2, 3, \ldots, p-1$), $\boldsymbol{x}(n)$ tends to the eigenvector $\boldsymbol{x}_j$.*
(iii) *For $\ell_{p-1} < \varepsilon < x_p$, $\boldsymbol{x}(n)$ tends to the eigenvector $\boldsymbol{x}_p$.*
(iv) *For $\ell_- < \varepsilon < 0$, $\boldsymbol{x}(n)$ tends to the eigenvector $\boldsymbol{x}_N$.*

**Proof.** (i) It suffices to show that for $x < \ell_-,\ 0 < x < \ell_1,\ x_p < x$,

$$\max_{j\neq 1} |f(x,\lambda_j)| < |f(x,\lambda_1)|. \tag{29}$$

For each $j$ $(j \neq 1)$, $|f(x,\lambda_j)| < |f(x,\lambda_1)|$ for $x > \beta_j^1$. Since $x_p$ is the maximum of $\{\beta_j^1\}$, (29) holds for $x > x_p$. We have $|f(x,\lambda_j)| < |f(x,\lambda_1)|$ for $0 < x < 2/(\lambda_1+\lambda_j)$. Since $\ell_1 = \min_{j\neq 1}\{2/(\lambda_1+\lambda_j)\}$, (29) holds for $0 < x < \ell_1$. We also have $|f(x,\lambda_j)| < |f(x,\lambda_1)|$ for $x < \alpha_j^1$, where $\alpha_j^1$ is the negative root of $f(x,\lambda_1) = -f(x,\lambda_j)$. Since $\ell_- = \min_{j\neq 1}\{\alpha_j^1\}$, (29) holds for $x < \ell_-$.

(ii) It is sufficient to show that for $\ell_{j-1} < x < \ell_j$ $(j = 2,3,\ldots,p-1)$,

$$\max_{k\neq j} |f(x,\lambda_k)| < |f(x,\lambda_j)|. \tag{30}$$

It holds that $|f(x,\lambda_1)| < |f(x,\lambda_j)|$ for $2/(\lambda_1+\lambda_j) < x < \beta_j^1$. Since $2/(\lambda_1+\lambda_j) \leqslant \ell_{j-1}$ and $\ell_j \leqslant \beta_j^1$ by Lemma 1, we have $|f(x,\lambda_1)| < |f(x,\lambda_j)|$ for $\ell_{j-1} < x < \ell_j$. We now suppose that $k \neq 1$ and $k \neq j$. If $x > (1+\sqrt{3})/\lambda_k$ and $\ell_{j-1} < x < \ell_j$, then $0 < f(x,\lambda_k) < f(x,\lambda_1)$. Using (28) in the proof of Lemma 1, if $0 < x \leqslant (1+\sqrt{3})/\lambda_k$ and $\ell_{j-1} < x < \ell_j$, then $f(x,\lambda_j) < f(x,\lambda_k) \leqslant 0$. Henceforth (30) holds for $\ell_{j-1} < x < \ell_j$.

(iii) The same discussion as in (ii) leads to the desired conclusion.

(iv) For each $j$ $(j \neq N)$, let $\alpha_j^N$ be a negative root of $f(x,\lambda_j) = -f(x,\lambda_N)$. Then $|f(x,\lambda_j)| < |f(x,\lambda_N)|$ for $\alpha_j^N < x < 0$. Since $\ell_-$ is the maximum of $\{\alpha_j^N\}$, $\max_{j\neq N}|f(x,\lambda_j)| < |f(x,\lambda_N)|$ holds for $\ell_- < x < 0$. □

Theorem 2 shows that by using difference equation (26) we can calculate the eigenvalues from $\lambda_1$ until $\lambda_p$, and $\lambda_N$. But the convergence is very slow except for $\lambda_1$. For small positive $\varepsilon$, $\langle \boldsymbol{x}(n), A\boldsymbol{x}(n)\rangle$ converges to $\lambda_1$, and for small negative $\varepsilon$, $\langle \boldsymbol{x}(n), A\boldsymbol{x}(n)\rangle$ converges to $\lambda_N$. This means that the difference equation (26) is germane to continuous equation (5) for small stepsize $\varepsilon$.

Next, we shall find out the optimal value $\varepsilon_{\text{opt}}$ for the sequence $\{\boldsymbol{x}(n)\}_{n=0,1,2,\ldots}$ defined in (26). In this case we suppose that the minimal eigenvalue is zero.

**Definition 2.** Let $\lambda_2 > \cdots > \lambda_{N-1} > \lambda_N \geqslant 0$. For $j = 3,4,\ldots,N$, we denote the two roots of the equation $f(x,\lambda_2) = -f(x,\lambda_j)$ by $\alpha_j^2 < 0 < \beta_j^2$, and define $\bar{x}_q \equiv \beta_q^2 = \max_{3\leqslant j\leqslant N}\{\beta_j^2\}$, where if there exist two integers $q$ and $q'$ such that $\beta_q^2 = \beta_{q'}^2$, then we adopt the larger one.

In a similar way as in Lemma 1, we obtain that $\ell_{q-1} \leqslant \bar{x}_q < \ell_q$ for $q = 3,4,\ldots,N$. It is easy to prove the following lemma.

**Lemma 2.** *Let $a > b \geqslant 0$. We denote the two roots of the equation $f(x,a) = -f(x,b)$ by $\alpha < 0 < \beta$. Then the function $g(x) \equiv |f(x,b)|/|f(x,a)|$ is strictly increasing for $(1-\sqrt{3})/b \leqslant x \leqslant \alpha$, $x \geqslant (1+\sqrt{3})/b$. And $g(x)$ is a strictly decreasing function for $x \leqslant (1-\sqrt{3})/b$, $\beta \leqslant x \leqslant (1+\sqrt{3})/b$. Here, when $b = 0$, we set $(1-\sqrt{3})/b = -\infty$, $(1+\sqrt{3})/b = +\infty$.*

**Theorem 3.** *Let $A$ be an $N \times N$ real symmetric matrix having eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_{N-1} > \lambda_N = 0$. Then the optimal $\varepsilon$ for the most rapid convergence of $\{\boldsymbol{x}(n)\}_{n=0,1,2,\ldots}$ defined in (26) to the*

*eigenvector* $\boldsymbol{x}_1$ *corresponding to* $\lambda_1$ *is given by*

$$\varepsilon_{\text{opt}} = \bar{x}_q, \tag{31}$$

*where* $\bar{x}_q$ *is as defined in Definition* 2.

**Proof.** We shall show that $\bar{x}_q$ is the optimal value in the following four intervals: $x < \ell_-$, $0 < x < \ell_1$, $x_p < x \leqslant \bar{x}_q$, $x \geqslant \bar{x}_q$, respectively.

(i) When $x \geqslant \bar{x}_q$, using Theorem 2 for $\lambda_2 > \lambda_3 > \cdots > \lambda_{N-1} > \lambda_N = 0$ instead of $\lambda_1 > \lambda_2 > \cdots > \lambda_{N-1} > \lambda_N \geqslant 0$, we obtain, for $x \geqslant \bar{x}_q$,

$$\max_{j \neq 1,2} |f(x, \lambda_j)| \leqslant |f(x, \lambda_2)|.$$

By Lemma 2, $g(x) = |f(x, \lambda_2)/f(x, \lambda_1)|$ is a strictly increasing function on $x \geqslant (1 + \sqrt{3})/\lambda_2$. Since obviously $\bar{x}_q \geqslant (1 + \sqrt{3})/\lambda_2$, we have

$$\left| \frac{f(\bar{x}_q, \lambda_2)}{f(\bar{x}_q, \lambda_1)} \right| = \min_{x \geqslant \bar{x}_q} \max_{j \neq 1} \left| \frac{f(x, \lambda_j)}{f(x, \lambda_1)} \right|. \tag{32}$$

(ii) When $x_p < x \leqslant \bar{x}_q$, using Theorem 2 for $\lambda_2 > \lambda_3 > \cdots > \lambda_{N-1} > \lambda_N = 0$, we have, for each $j = p, p+1, \ldots, q-1$, and for $\ell_{j-1} \leqslant x \leqslant \ell_j$, $\max_{k \neq 1} |f(x, \lambda_k)| \leqslant |f(x, \lambda_j)|$. Also we have, for $\ell_{q-1} \leqslant x \leqslant \bar{x}_q$, $\max_{k \neq 1} |f(x, \lambda_k)| \leqslant |f(x, \lambda_q)|$. By Lemma 2, $g(x) = |f(x, \lambda_p)/f(x, \lambda_1)|$ is a strictly decreasing function on $x_p \leqslant x \leqslant \ell_p$, because $\ell_p \leqslant (1 + \sqrt{3})/\lambda_p$, so the minimum is obtained at $x = \ell_p$. And $g_1(x) = |f(x, \lambda_{p+1})/f(x, \lambda_1)|$ is also a strictly decreasing function on $\ell_p \leqslant x \leqslant \ell_{p+1}$, so the minimum is attained at $x = \ell_{p+1}$, and so on. Hence we have

$$\left| \frac{f(\bar{x}_q, \lambda_2)}{f(\bar{x}_q, \lambda_1)} \right| = \left| \frac{f(\bar{x}_q, \lambda_q)}{f(\bar{x}_q, \lambda_1)} \right| = \min_{x_p < x \leqslant \bar{x}_q} \max_{j \neq 1} \left| \frac{f(x, \lambda_j)}{f(x, \lambda_1)} \right|. \tag{33}$$

(iii) When $x < \ell_-$, let $\alpha$ be the negative root of $f(x, \lambda_2) = -f(x, \lambda_N)$ and $\beta$ be the positive root of $f(x, \lambda_2) = 3$, that is, $\alpha = (1 - \sqrt{5})/\lambda_2$, $\beta = (1 + \sqrt{6})/\lambda_2$. Here notice that $\max_x \{-f(x, \lambda_q)\} = 3$, hence we have $\bar{x}_q \leqslant \beta$. It follows that

$$\left| \frac{f(\bar{x}_q, \lambda_2)}{f(\bar{x}_q, \lambda_1)} \right| \leqslant \left| \frac{f(\beta, \lambda_2)}{f(\beta, \lambda_1)} \right| = \frac{3}{f(\beta, \lambda_1)} < \frac{2}{f(\alpha, \lambda_1)} = \left| \frac{f(\alpha, \lambda_2)}{f(\alpha, \lambda_1)} \right|$$

$$= \min_{x < \ell_-} \max_{j \neq 1} \left| \frac{f(x, \lambda_j)}{f(x, \lambda_1)} \right|,$$

where the second inequality is obtained by

$$2f(\beta, \lambda_1) - 3f(\alpha, \lambda_1) = (6\sqrt{5} + 4\sqrt{6} - 4)r^2 - (6\sqrt{5} + 4\sqrt{6} - 2)r + 2 > 0,$$

for $r \equiv \lambda_1/\lambda_2 > 1$, and the last equality is obtained by Lemma 2.

(iv) When $0 < x < \ell_1$, it holds that

$$0 < -f(x, \lambda_2) < -f(x, \lambda_1) \quad \text{for } 0 < x < \ell_1.$$

We set $g(x) = f(x,\lambda_2)/f(x,\lambda_1)$. Since $g(0) = g(\ell_1) = 1$, the minimum of $g(x)$ in $0 < x < \ell_1$ is obtained at $0 < \gamma < \ell_1$ such that $g'(\gamma) = 0$. We shall show that

$$\frac{f(\beta,\lambda_2)}{f(\beta,\lambda_1)} < \frac{f(\gamma,\lambda_2)}{f(\gamma,\lambda_1)}, \tag{34}$$

where $\beta = (1+\sqrt{6})/\lambda_2$, which is the same as in (iii). Inequality (34) is obtained as follows:

$$f(\beta,\lambda_1)f(\gamma,\lambda_2) - f(\beta,\lambda_2)f(\gamma,\lambda_1) = 2(\lambda_1-\lambda_2)(\gamma-\beta)\{\lambda_1\lambda_2\beta\gamma + (\lambda_1+\lambda_2)(\beta+\gamma) - 2\} < 0,$$

because $\gamma < \beta$ and $\beta\lambda_2 = 1+\sqrt{6}$. Here we note that $f(\beta,\lambda_1) > 0$ and $f(\gamma,\lambda_1) < 0$. □

Finally, we demonstrate that this convergence is faster than Wilkinson's method. When $\lambda_N = 0$, the convergence rate of Wilkinson's two-step method is $\{\lambda_2/(2\lambda_1-\lambda_2)\}^2$.

**Theorem 4.** *Let $A$ be an $N\times N$ real symmetric matrix having eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_{N-1} > \lambda_N = 0$. Suppose that $\lambda_1 < (25+10\sqrt{6})\lambda_2$. Then it follows that*

$$\left|\frac{f(\bar{x}_q,\lambda_2)}{f(\bar{x}_q,\lambda_1)}\right| < \left(\frac{\lambda_2}{2\lambda_1-\lambda_2}\right)^2, \tag{35}$$

*where $\bar{x}_q$ is as defined in Definition 2.*

**Proof.** Let $\beta$ be the positive root of $f(x,\lambda_2) = 3$, that is, $\beta = (1+\sqrt{6})/\lambda_2$. Then it follows that

$$\left|\frac{f(\bar{x}_q,\lambda_2)}{f(\bar{x}_q,\lambda_1)}\right| \leqslant \left|\frac{f(\beta,\lambda_2)}{f(\beta,\lambda_1)}\right| = \frac{3}{\beta^2\lambda_1^2 - 2\beta\lambda_1 - 2} < \frac{(\lambda_2)^2}{(2\lambda_1-\lambda_2)^2},$$

where the first inequality is obtained by Theorem 3, and the last inequality is obtained by the fact that

$$\frac{3}{(1+\sqrt{6})^2r^2 - 2(1+\sqrt{6})r - 2} < \frac{1}{(2r-1)^2}, \tag{36}$$

for $1 < r = \lambda_1/\lambda_2 < 5(5+2\sqrt{6})$. □

## 6. A new algorithm

For real numbers $k > 0$, we consider the following difference equation which is similar to (26):

$$\boldsymbol{x}(n+1) = \frac{(I+\varepsilon A - (\varepsilon^2/2k)A^2)\boldsymbol{x}(n)}{\|(I+\varepsilon A - (\varepsilon^2/2k)A^2)\boldsymbol{x}(n)\|}, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{37}$$

Now, we define

$$f(x,a;k) \equiv a^2x^2 - 2kax - 2k. \tag{38}$$

Let $\beta(k)$ be the positive root of $f(x,\lambda_2;k) = k^2+2k$, that is,

$$\beta(k) = \frac{k+\sqrt{2k^2+4k}}{\lambda_2}.$$

We notice that $\max_x\{-f(x,\lambda_j;k)\} = k^2+2k$ for any $\lambda_j \neq 0$.

**Theorem 5.** *Let $A$ be an $N \times N$ real symmetric matrix having eigenvalues $\lambda_1 > \lambda_2 \geqslant \cdots \geqslant \lambda_{N-1} > \lambda_N = 0$. Then the sequence $\boldsymbol{x}(n)$ defined by (37) for $\varepsilon = \beta(k)$ converges to the eigenvector $\boldsymbol{x}_1$ as $n \to \infty$, where $\beta(k)$ is the positive root of $f(x, \lambda_2; k) = k^2 + 2k$.*

**Proof.** It suffices to show that

$$\max_{j \neq 1} |f(\beta(k), \lambda_j; k)| < |f(\beta(k), \lambda_1; k)|. \tag{39}$$

Let $\beta_j^1(k)$ be the positive root of $f(x, \lambda_1; k) = -f(x, \lambda_j; k)$, where $j = 2, \ldots, N$. And let $\beta_1(k)$ be the positive root of $f(x, \lambda_1; k) = k^2 + 2k$. Since

$$-f(x, \lambda_j; k) \leqslant k^2 + 2k = f(\beta_1(k), \lambda_1; k),$$

we obtain $\beta_j^1(k) \leqslant \beta_1(k)$. It is clear that $\beta_1(k) \leqslant \beta(k)$. For each $j$ $(j \neq 1)$, $|f(x, \lambda_j; k)| < |f(x, \lambda_1; k)|$, where $x > \beta_j^1(k)$. Since $\beta_j^1(k) \leqslant \beta(k)$, (39) holds. □

Convergence rate of the above algorithm is $(k^2 + 2k)/f(\beta(k), \lambda_1; k)$, which yields the following lemma:

**Lemma 3.** *Let*

$$g(k) \equiv \frac{k^2 + 2k}{f(\beta(k), \lambda_1; k)},$$

*where $\beta(k)$ is the positive root of $f(x, \lambda_2; k) = k^2 + 2k$. Then $g(k)$ is a strictly decreasing function for $k > 0$.*

**Proof.** We have

$$g(k) = \frac{k+2}{f(\beta(k), \lambda_1; k)/k} = \frac{k+2}{2r(r-1)(k + \sqrt{2k^2 + 4k}) + r^2(k+4) - 2},$$

where $r = \lambda_1/\lambda_2 > 1$. It follows that

$$g'(k) = \frac{-2(r-1)\{(r-1)\sqrt{2k^2 + 4k} + 2r(k+2)\}}{\{f(\beta(k), \lambda_1; k)/k\}^2 \sqrt{2k^2 + 4k}} < 0. \qquad \square$$

**Remark 2.** The condition such as $\lambda_1 < (25 + 10\sqrt{6})\lambda_2$ in Theorem 4 is not necessary for $k > (8\sqrt{2} - 4)/7$.

By Lemma 3, the convergence becomes more rapid as $k \to \infty$. So in (37), for $\varepsilon = \beta(k)$, if we let $k \to \infty$, then a new algorithm is obtained:

$$\boldsymbol{x}(n+1) = \frac{(A - \varepsilon_0 A^2)\boldsymbol{x}(n)}{\|(A - \varepsilon_0 A^2)\boldsymbol{x}(n)\|}, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{40}$$

where

$$\varepsilon_0 = \lim_{k \to \infty} \frac{\beta(k)}{2k} = \lim_{k \to \infty} \frac{k + \sqrt{2k^2 + 4k}}{2k\lambda_2} = \frac{1 + \sqrt{2}}{2\lambda_2}.$$

The rate of convergence is given by

$$\lim_{k\to\infty}\frac{f(\beta(k),\lambda_2;k)}{f(\beta(k),\lambda_1;k)}=\frac{1}{(3+2\sqrt{2})r^2-2(1+\sqrt{2})r}.$$

**Numerical example 1.** Consider the matrix

$$A=\frac{1}{7}\begin{pmatrix}488 & -95 & 296 & 5\\ -95 & 197 & -7 & -292\\ 296 & -7 & 188 & -101\\ 5 & -292 & -101 & 485\end{pmatrix}$$

for which $\lambda_1=98,\ \lambda_2=97,\ \lambda_3=1,\ \lambda_4=-2$. Let $\boldsymbol{x}(0)=(0.1,2.8,1,2.3)^{\mathrm{T}}$ be the initial vector.

(i) Our method

Apply the algorithm (2) to $B\equiv A-\lambda_4 I$ and $\varepsilon_0\equiv(1+\sqrt{2})/2(\lambda_2-\lambda_4)$. Then for $n=60$ (actually 120 steps are needed), we have an approximate largest eigenvalue $\langle\boldsymbol{x}(60),B\boldsymbol{x}(60)\rangle+\lambda_4=\langle\boldsymbol{x}(60),A\boldsymbol{x}(60)\rangle=97.9997$. For $n=125$, we have $\lambda_1=97.9999999501$. The rate of convergence is $1/\{(3+2\sqrt{2})r^2-2(1+\sqrt{2})r\}=0.934956$, where $r\equiv(\lambda_1-\lambda_4)/(\lambda_2-\lambda_4)=100/99$. We usually do not know the exact eigenvalues $\lambda_2,\lambda_4$ beforehand, but we can still use our method for the approximate eigenvalues. For example, let $\lambda_2'=96.8,\ \lambda_4'=-1.7$ be the approximate eigenvalues corresponding to $\lambda_2$, $\lambda_4$. Apply the algorithm (2) to $B'\equiv A-\lambda_4' I$ and $\varepsilon_0'\equiv(1+\sqrt{2})/2(\lambda_2'-\lambda_4')$. For $n=60$, we have an approximate largest eigenvalue $\langle\boldsymbol{x}(60),B'\boldsymbol{x}(60)\rangle+\lambda_4'=97.9997$. For $n=125$, we have $\lambda_1=97.9999999457$.

(ii) The power method with a shift of the origin (Wilkinson's method)

Compute the largest eigenvalue using (1). For $n=120$, we have $\lambda_1=53.0577$. This convergence is very slow, for the reason noted below. The rate of convergence is $1/(2r-1)^2=0.960788$. Wilkinson [6] remarked that before using his method, the power method in 5 or 6 iterations should be applied. So after applying the power method in 6 iterations, compute the largest eigenvalue using (1). For $n=114$, we have $\lambda_1=97.9908$. For $n=414$, we have $\lambda_1=97.9999999432$. Using the approximate eigenvalues $\lambda_2'$, $\lambda_4'$, after applying the power method in 6 iterations, for $n=114$, Wilkinson's method yields $\lambda_1=97.9909$. For $n=414$, it yields $\lambda_1=97.9999999441$.

(iii) The power method

For $n=120$, we have an approximate largest eigenvalue $\lambda_1=97.9214$. For $n=800$, we have $\lambda_1=97.9999999253$. The rate of convergence is $(\lambda_2/\lambda_1)^2=0.979696$.

We note that the convergence is depending upon the initial vector. In the above example 1, using the notation $\boldsymbol{x}_0=\sum_{j=1}^{4}\alpha_j\boldsymbol{x}_j$ in Section 1, the coefficients $\alpha_j$ are taken such that $\alpha_1=\alpha_2=0.1$, $\alpha_3=\alpha_4=1$. If we use the power method, the terms of $\boldsymbol{x}_3$ and $\boldsymbol{x}_4$ will rapidly die out. But they still remain in Wilkinson's method. That is why the convergence is very slow. Our method involves the same situation. Let $\lambda_1>\lambda_2\geqslant x>\lambda_N=0$, then $\max_x|x-\varepsilon_0x^2|/|\lambda_1-\varepsilon_0\lambda_1^2|$ is obtained at $x=\lambda_2$ or $x=1/\varepsilon_1\equiv 1/2\varepsilon_0=\lambda_2(\sqrt{2}-1)$. If there exist the eigenvalues near to $1/\varepsilon_1$, then the convergence will be slow. For example, let $\lambda_{\mathrm{m}}$ denote such an eigenvalue and $\alpha_{\mathrm{m}}\boldsymbol{x}_{\mathrm{m}}$ is not small in the initial vector, then the convergence becomes slow. As Wilkinson's method, we apply the following algorithm in 5 or 6 iterations before using our method,

$$\boldsymbol{x}(n+1)=\frac{(I-\varepsilon_1 A)\boldsymbol{x}(n)}{\|(I-\varepsilon_1 A)\boldsymbol{x}(n)\|},\qquad \boldsymbol{x}(0)=\boldsymbol{x}_0, \tag{41}$$

where $\varepsilon_1\equiv 1/\lambda_2(\sqrt{2}-1)$. The next example describes that case.

**Numerical example 2.** Consider the matrix

$$A=\frac{1}{7}\begin{pmatrix} 525 & -21 & 259 & 42 \\ -21 & 345 & -81 & -218 \\ 259 & -81 & 225 & -138 \\ 42 & -218 & -138 & 522 \end{pmatrix}$$

for which $\lambda_1=98,\ \lambda_2=97,\ \lambda_3=38,\ \lambda_4=-2$. Let $\boldsymbol{x}(0)=(0.8,1.9,-0.7,1.7)^{\mathrm{T}}$ be the initial vector $(\alpha_1=0.1,\alpha_2=\alpha_4=0.2,\alpha_3=1)$.

(i) Our method

Apply the algorithm (2) to $B\equiv A-\lambda_4 I$ and $\varepsilon_0\equiv(1+\sqrt{2})/2(\lambda_2-\lambda_4)$, then for $n=60$, we have $\langle\boldsymbol{x}(60),A\boldsymbol{x}(60)\rangle=96.3057$. This convergence is slow, so we apply the algorithm (41) to $B=A-\lambda_4 I$ and $1/\varepsilon_1=(\lambda_2-\lambda_4)(\sqrt{2}-1)$ in 6 iterations, after that we use algorithm (2), then for $n=57$, we have $\lambda_1=97.9985$. For $n=122$, we have $\lambda_1=97.9999997568$.

(ii) Wilkinson's method

Compute the largest eigenvalue using (1), for $n=120$, we have $\lambda_1=94.881$. After the power method in 6 iterations, using (1), for $n=114$, we have $\lambda_1=97.9643$. For $n=410$, we have $\lambda_1=97.9999997334$.

(iii) The power method

When $n=120$, we have an approximate largest eigenvalue $\lambda_1=97.7456$. When $n=800$, we have $\lambda_1=97.9999997014$.

**Remark 3.** For example, in the case of $\lambda_1=51,\ \lambda_2=50,\ \lambda_3=-46,\ \lambda_4=-49$ and $\alpha_1=\alpha_2=0.1,\ \alpha_3=\alpha_4=1$, if we apply the power method before using Wilkinson's method, the terms of $\boldsymbol{x}_3$ and $\boldsymbol{x}_4$ will not die out. But our method is the same as numerical example 1.

Returning to our discussion, if we know the algorithm in advance

$$\boldsymbol{x}(n+1)=\frac{(A-\varepsilon A^2)\boldsymbol{x}(n)}{\|(A-\varepsilon A^2)\boldsymbol{x}(n)\|},\quad \boldsymbol{x}(0)=\boldsymbol{x}_0, \tag{42}$$

then $\varepsilon_0$ can be easily obtained by another way. Let $f(x,a)=a-a^2x$. In general, for $a>b\geqslant 0$, the equation $|f(x,a)|=|f(x,b)|$ has two roots $\alpha=1/(a+b)$ and $\beta=(a+b)/(a^2+b^2)$. If $x\geqslant\beta$, then $|f(x,b)|\leqslant|f(x,a)|$. For fixed $a>0$, and for any $b$ such that $a>b\geqslant 0$, we have

$$\frac{1}{a}\leqslant\beta(b)\leqslant\frac{1+\sqrt{2}}{2a},$$

where $\beta(b)\equiv(a+b)/(a^2+b^2)$. We note that the maximum of $\beta(b)$ is obtained at $b=a(\sqrt{2}-1)$. Hence, in the case of $a=\lambda_2,\ b=\lambda_j\ (j=3,4,\ldots,N)$, we have $|f(\varepsilon,\lambda_j)|\leqslant|f(\varepsilon,\lambda_2)|$ for $\varepsilon\geqslant\varepsilon_0\equiv(1+\sqrt{2})/2\lambda_2$. Observing that the function $g(x)\equiv|f(x,\lambda_2)|/|f(x,\lambda_1)|$ is strictly increasing for $x\geqslant\varepsilon_0$, it follows that

$$\min_{\varepsilon\geqslant\varepsilon_0}\max_{j\neq 1}\left|\frac{f(\varepsilon,\lambda_j)}{f(\varepsilon,\lambda_1)}\right|=\left|\frac{f(\varepsilon_0,\lambda_2)}{f(\varepsilon_0,\lambda_1)}\right|. \tag{43}$$

For algorithm (42) $\varepsilon_0$ is not an optimum value. In order to compute the optimum value $\varepsilon_{\mathrm{opt}}$, we need all eigenvalues of $A$. The next lemma shows that under some conditions $\varepsilon_0$ equals to $\varepsilon_{\mathrm{opt}}$.

**Lemma 4.** *Let $A$ be an $N\times N$ real symmetric matrix having eigenvalues $\lambda_1>\lambda_2\geqslant\cdots\geqslant\lambda_{N-1}>\lambda_N=0$. We suppose that there exists $\lambda_m$ $(m=3,4,\ldots,N-1)$ such that $\lambda_m=\lambda_2(\sqrt{2}-1)$. Then $\varepsilon_0=(1+\sqrt{2})/2\lambda_2$ is the optimum value for the algorithm* (42).

**Proof.** Let $f(x,a)=a-a^2x$. We define $h(x)=\max_{j\neq1}|f(x,\lambda_j)|$. Let $\alpha<\beta$ be the two roots of $|f(x,\lambda_1)|=h(x)$. Since $h(x)\geqslant|f(x,\lambda_1)|$ for $\alpha\leqslant x\leqslant\beta$, we have

$$\min_x\max_{j\neq1}\left|\frac{f(x,\lambda_j)}{f(x,\lambda_1)}\right|=\min_{x<\alpha,x>\beta}\max_{j\neq1}\left|\frac{f(x,\lambda_j)}{f(x,\lambda_1)}\right|. \tag{44}$$

For each $j=3,4,\ldots,N$, we have $|f(x,\lambda_j)|\leqslant|f(x,\lambda_2)|$ for $x\leqslant1/(\lambda_2+\lambda_j)$, $x\geqslant(\lambda_2+\lambda_j)/(\lambda_2^2+\lambda_j^2)$. By the assumption on $\lambda_m$, we have $\max_{j\neq1,2}\{(\lambda_2+\lambda_j)/(\lambda_2^2+\lambda_j^2)\}=(\lambda_2+\lambda_m)/(\lambda_2^2+\lambda_m^2)=\varepsilon_0$. So we obtain $h(x)=|f(x,\lambda_2)|=-f(x,\lambda_2)$ for $x\geqslant\varepsilon_0$. We note that $\alpha=1/(\lambda_1+\lambda_2)$. We have $h(x)=f(x,\lambda_2)$ for $x<\alpha$. For each $\beta<x\leqslant\varepsilon_0$, there exists $j$ $(j=2,3,\ldots,N)$ such that $h(x)=|f(x,\lambda_j)|=f(x,\lambda_j)$. We define $g(x)=h(x)/|f(x,\lambda_1)|$. Then $g(x)=f(x,\lambda_2)/f(x,\lambda_1)$ for $x<\alpha$, $x\geqslant\varepsilon_0$, and for each $\beta<x\leqslant\varepsilon_0$, there exists $j$ $(j=2,3,\ldots,N)$ such that $g(x)=-f(x,\lambda_j)/f(x,\lambda_1)$. Since, for $g(x)=f(x,\lambda_2)/f(x,\lambda_1)$, $g'(x)=\lambda_1\lambda_2(\lambda_1-\lambda_2)/(\lambda_1-\lambda_1^2x)^2>0$, the function $g(x)$ is strictly increasing for $x<\alpha$, $x\geqslant\varepsilon_0$. Similarly, $g(x)$ is a strictly decreasing function for $\beta<x\leqslant\varepsilon_0$. Since $\lim_{x\to-\infty}g(x)=(\lambda_2/\lambda_1)^2=1/r^2$, the minimum of $g(x)$ is obtained at $x=\varepsilon_0$. □

## 7. Discretization by matrix Riccati equation

In this section it will be shown that (7) can also be discretized by the matrix Riccati equation. When $Y,A_1,B_1,C_1,D_1$ are $N\times N$ real matrices, we consider the following matrix Riccati equation:

$$\frac{\mathrm{d}Y(t)}{\mathrm{d}t}=A_1+YB_1+C_1Y+YD_1Y,\qquad Y(0)=Y_0. \tag{45}$$

It is well known that the solution of this matrix Riccati equation is given by $Y=VW^{-1}$, where $V(t),W(t)$ are the solutions of the following system of linear differential equations:

$$\begin{pmatrix}\dot V\\ \dot W\end{pmatrix}=\begin{pmatrix}C_1 & A_1\\ -D_1 & -B_1\end{pmatrix}\begin{pmatrix}V\\ W\end{pmatrix},\qquad\begin{pmatrix}V(0)\\ W(0)\end{pmatrix}=\begin{pmatrix}Y_0\\ I\end{pmatrix},$$

where $\dot V=\mathrm{d}V/\mathrm{d}t$, $\dot W=\mathrm{d}W/\mathrm{d}t$. Eq. (7) can be transformed into the following matrix Riccati equation:

$$\dot Y=DY-YDY$$

with the linear equations

$$\dot V=DV,\qquad V(0)=Y_0, \tag{46}$$

$$\dot W=DV,\qquad W(0)=I, \tag{47}$$

where

$$Y=\begin{pmatrix}y_1 & \cdots & y_1\\ \vdots & \ddots & \vdots\\ y_N & \cdots & y_N\end{pmatrix},\qquad V=\begin{pmatrix}v_1 & \cdots & v_1\\ \vdots & \ddots & \vdots\\ v_N & \cdots & v_N\end{pmatrix},$$

$$Y_0 = \begin{pmatrix} y_1(0) & \cdots & y_1(0) \\ \vdots & \ddots & \vdots \\ y_N(0) & \cdots & y_N(0) \end{pmatrix}.$$

From $\dot{W} = \dot{V}$, we have the solution $W(t) = I + V(t) - Y_0$ for $W$. Hence, we discretize (46) only. If we discretize (46) by the forward Euler method, then

$$V(1) - V(0) = \varepsilon DV(0). \tag{48}$$

Using $W(1) = I + V(1) - Y_0$, it follows that

$$V(1)W(1) = V(1) + V(1)V(1) - V(1)Y_0.$$

Notice that $V(1)V(1) = (\sum_{k=1}^{N} v_k)V(1)$ and $V(1)Y_0 = (\sum_{k=1}^{N} y_k(0))V(1)$. This gives us

$$V(1)W(1) = \alpha V(1),$$

where $\alpha = 1 + \sum_{k=1}^{N}(v_k(1) - y_k(0))$. Hence it follows that

$$Y(1) = V(1)W(1)^{-1} = \frac{1}{\alpha} V(1) = \frac{1}{\alpha}(I + \varepsilon D)Y_0.$$

From $\sum_{k=1}^{N} y_k(0) = 1$ and (48), we have $\alpha = \sum_{k=1}^{N} v_k(1) = \sum_{k=1}^{N}(1 + \varepsilon\lambda_k y_k(0))$. This yields

$$y_j(1) = \frac{(1 + \varepsilon\lambda_j)y_j(0)}{\sum_{k=1}^{N}(1 + \varepsilon\lambda_k)y_k(0)}.$$

With the same discussion as in Section 4, we have the following power method with a shift of origin:

$$\boldsymbol{x}(n+1) = \frac{(I + \varepsilon A)\boldsymbol{x}(n)}{\|(I + \varepsilon A)\boldsymbol{x}(n)\|}. \tag{49}$$

Similarly, if we discretize (46) by the backward Euler method, then the inverse iteration method is obtained. Also if we discretize (46) by the second-order forward Runge–Kutta method, then the following difference equation is obtained:

$$\boldsymbol{x}(n+1) = \frac{(I + \varepsilon A + (\varepsilon^2/2)A^2)\boldsymbol{x}(n)}{\|(I + \varepsilon A + (\varepsilon^2/2)A^2)\boldsymbol{x}(n)\|}, \quad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{50}$$

Comparing (26) and (50), the latter is a better approximation for the exact solution, but we could not obtain more noteworthy algorithm from this.

## Acknowledgements

The author would like to thank Professor Y. Nakamura and Professor R. Hirota for their useful information provided through personal communications. The author would also like to thank Professor M.T. Nakao for useful advice.

## References

[1] G.H. Golub, C.H.V. Loan, Matrix Computations, Johns Hopkins, Baltimore, MD, 1983.
[2] R. Hirota, S. Tsujimoto, T. Imai, Difference scheme of soliton equations, in: P.L. Christiansen, J.C. Eilbeck, R.D. Parmentier (Eds.), Future Directions of Nonlinear Dynamics in Physical and Biological Systems, Plenum, New York, 1993, pp. 7–15.
[3] Y. Nakamura, K. Kajiwara, H. Shiotani, On an integrable discretization of the Rayleigh quotient gradient system and the power method with a shift, J. Comput. Appl. Math. 96 (1998) 77–90.
[4] Y. Nakamura, Jacobi algorithm for symmetric eigenvalue problem and integrable gradient system of Lax form, Jpn J. Ind. Appl. Math. 14 (1997) 159–168.
[5] N. Shibutani, System of ordinary differential equations associated with quadratic nonlinear terms, Bull. Kurume Inst. of Technol. 11 (1987) 1–7.
[6] J.H. Wilkinson, The Algebraic Eigenvalue Problem, Oxford Univ. Press, Oxford, 1996, first published in 1965.